

S2dverification + GitLab

D. Manubens, N. Manubens, O. Mula

Outline

- SCRUM development model
- GIT branching + testing
 - Currently
 - New suggestion
- Pending tests
- Questions, comments, ...
- GitLab practical session

SCRUM development model

- Affects only maintenance tasks, not personal developments.
- The development happens in a sequence of stages: sprints.
- Product backlog and sprint backlogs here:

<https://docs.google.com/spreadsheets/d/16-7CvRxNEcpX4k25sYjcV3G4XoqYFwvGXn1W0fv3CAQ/edit?usp=sharing>

GIT branching + testing

- Currently:

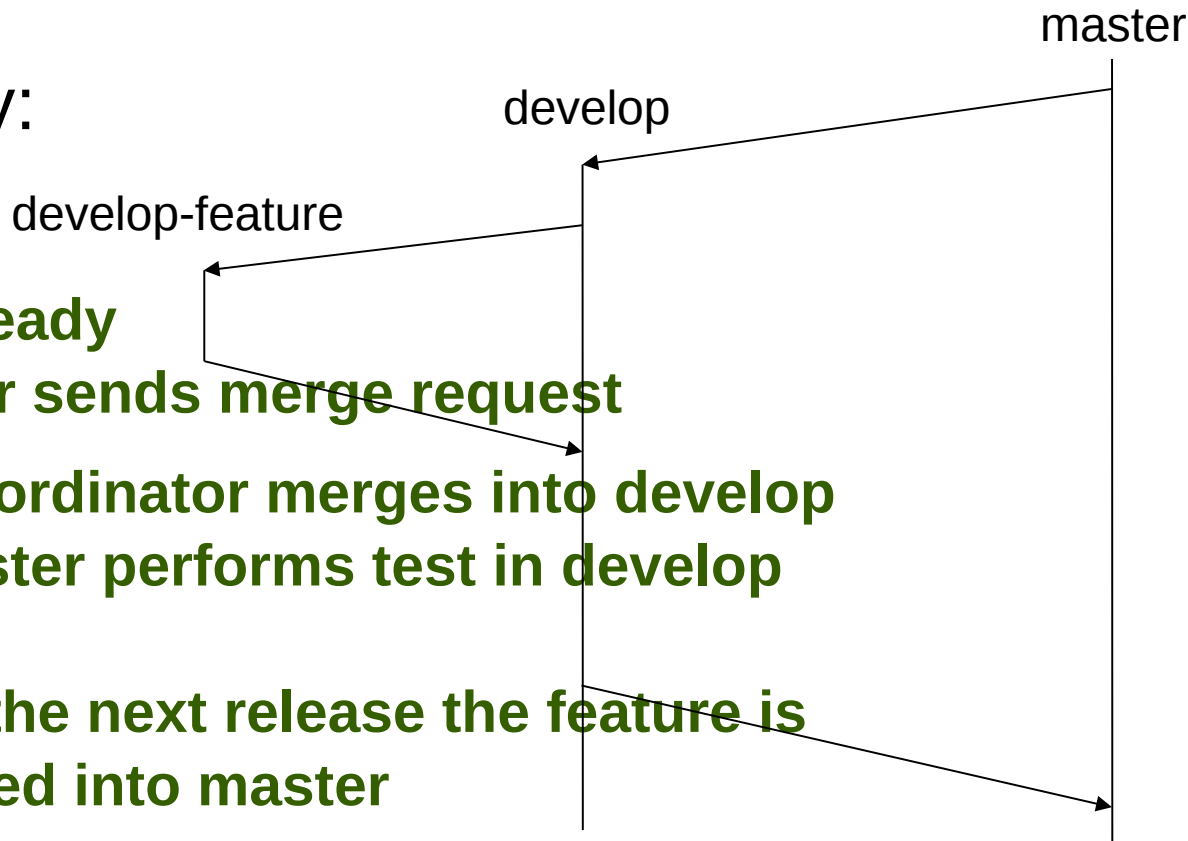
1- Feature ready

2- Developer sends merge request

3- Coordinator merges into develop

4- Tester performs test in develop

5- In the next release the feature is merged into master

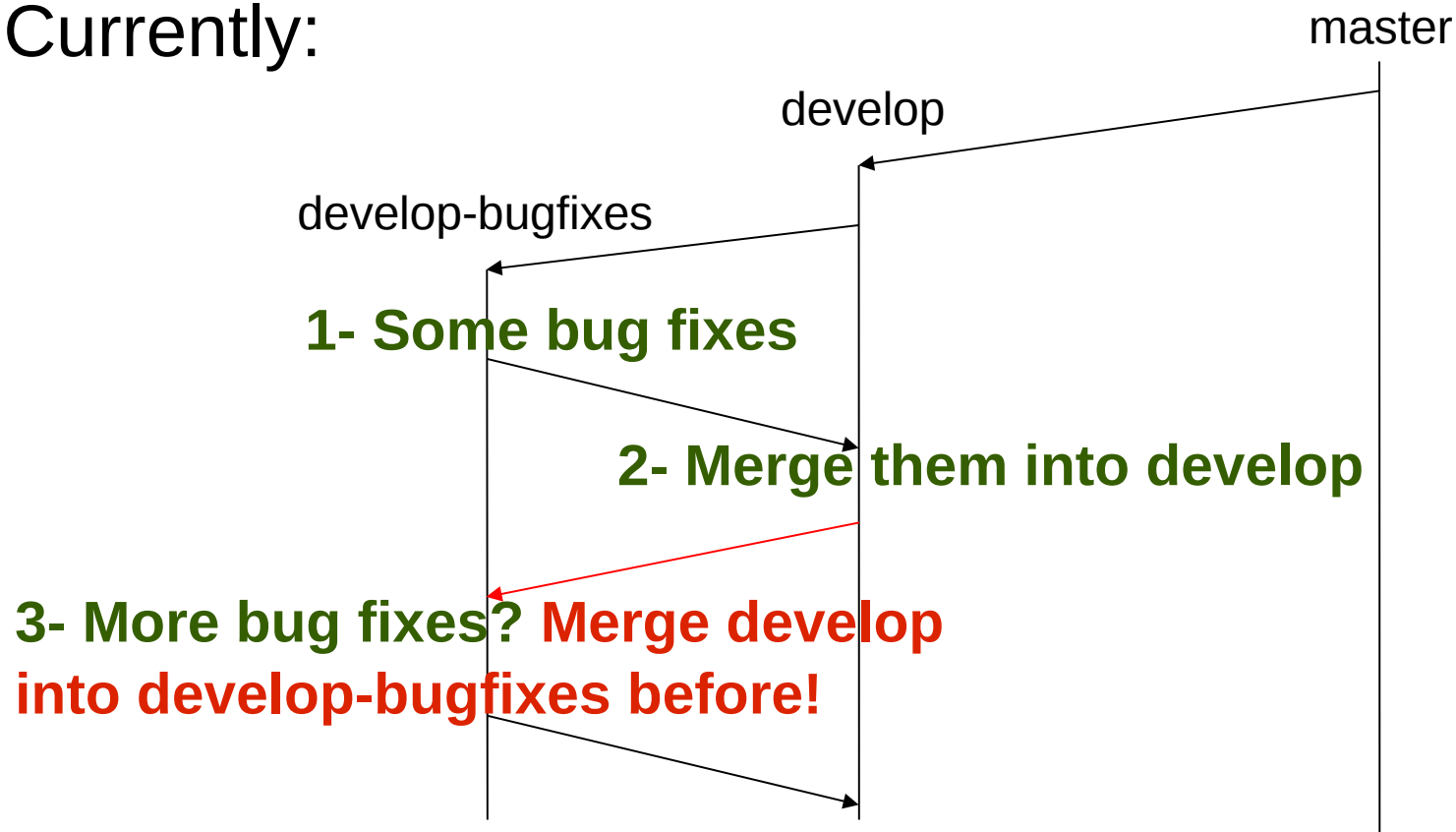


- Features untested in develop, may cause bugs

- Hard to release because everything must be tested

GIT branching + testing

- Currently:



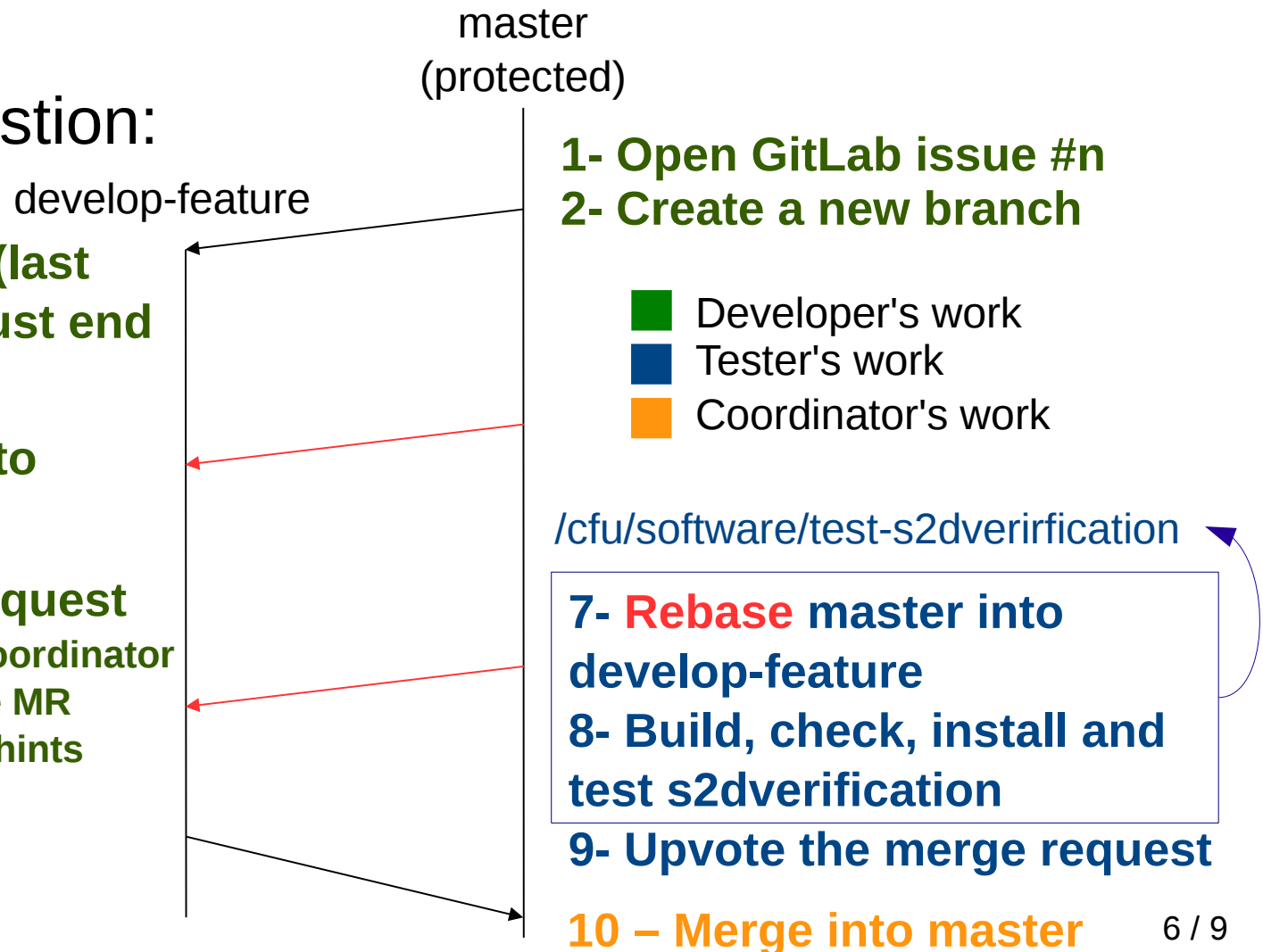
GIT branching + testing

- New suggestion:

3- Feature commits (last commit message must end with ' Fixes #n')

5- Rebase master into develop-feature

6- Create a merge request
- Assign the MR to the coordinator
- Put testers as /cc in the MR message and give them hints



GIT branching + testing

- New suggestion:
 - No risk of bugs in the main branch
 - Always releaseable
 - develop-bugfixes disappears
 - Do you agree?
 - Any suggestion?

Pending tests

Branch name	Testers
develop-plotStereoMaps	Neven + Danila
develop-loadIce	Neven
develop-loadObsOnly	Neven
develop-plotACC	Chloé
develop-ACC_permutation	Virginie
develop-corr-level-signif	(Chloé +) Virginie
develop-loadGlorys	Chloé

Questions, comments...

- Any new needs?
- Are you developing in s2dverification?
- ...

GitLab practical session

- Example of how to develop (live)